

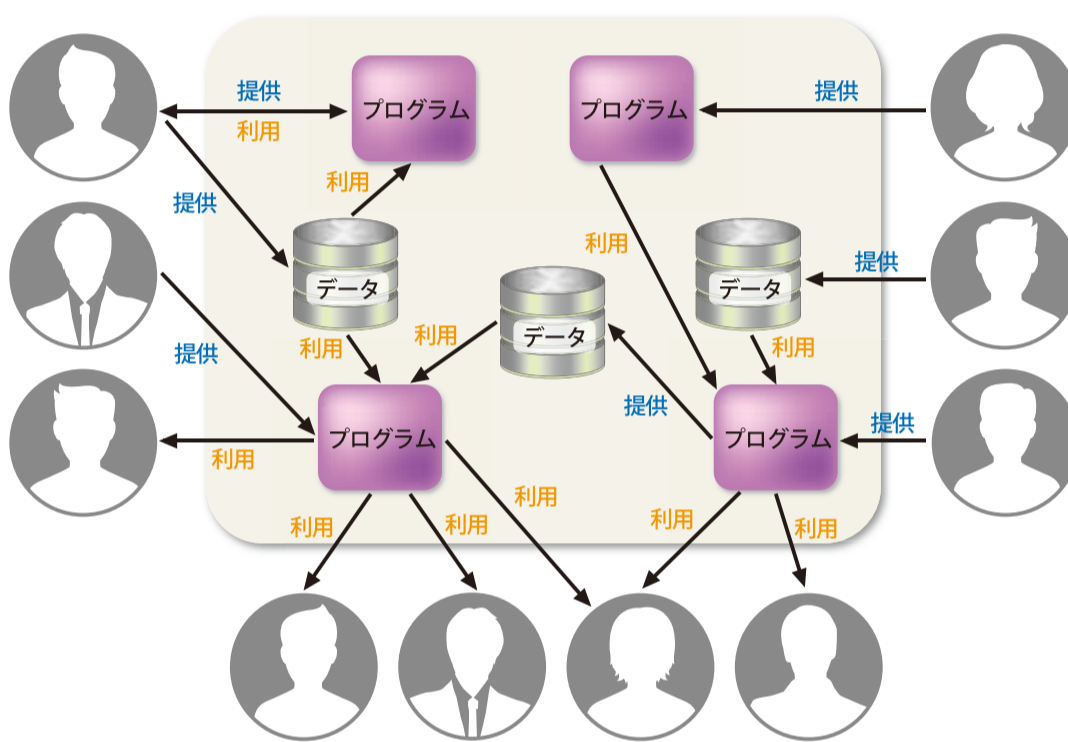
私たちの大きな目標は、「京」をもっと多くのユーザーに使ってもらうことです。そのために、プログラム開発者がプログラムを、データ提供者がデータを持ち寄る「計算ポータル」を構築し、プログラムとデータをさまざまな組み合わせで多くのユーザーに使ってもらえるしくみを提供したいと考えています。また計算ポータルの開発と同時に、2つの基盤技術を研究しています。1つは、仮想化技術を用いて「京」の中に仮想的な「京」を何台もつくる研究で、もう1つは、まちがったプログラムを事前に検証する方法の研究です。これらにより、多くの人に計算ポータルを公平かつ安全に利用してもらうことが可能になります。



チームリーダー
前田俊行 (まえだ・としゆき)

計算ポータルの構築

より多くのユーザーが「京」を使えるようにするには、プログラム開発者がプログラムを、データ提供者がデータを持ち寄る場が必要です。その場として、計算ポータルを構築します。

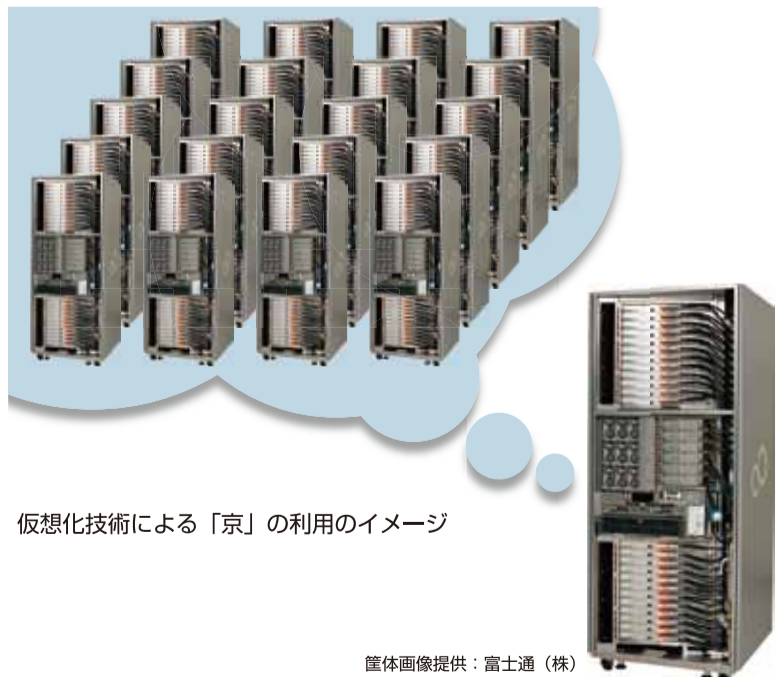


計算ポータルのイメージ

より多くの人に、より簡単に「京」を使ってもらうために

仮想化技術の研究

仮想化技術により、本物の「京」の中に仮想的な「京」を何台もつくることができます。これにより、ユーザーは公平に「京」を利用でき、また、データがほかのユーザーに漏れにくくなります。



仮想化技術による「京」の利用のイメージ

筐体画像提供：富士通（株）

プログラム検証技術の研究

計算ポータルにまちがったプログラムが置かれると、まちがいの影響が多くのユーザーに広まってしまいます。このため、計算ポータルに置くプログラムを事前に検証する必要があります。

$$\begin{array}{l}
 \frac{\Delta; C \vdash \Sigma = \Sigma' \otimes (\Gamma(r_d) \mapsto (\dots, \sigma_n, \dots))}{\Delta; \Gamma\{r_d \mapsto \sigma_n\}; C; \Sigma \vdash I} \quad \text{(LOAD)} \\
 \frac{\Delta; C \vdash \Sigma = \Sigma' \otimes (\Gamma(r_d) \mapsto (\dots, \sigma_n, \dots))}{\Delta; \Gamma; C; \Sigma' \otimes \{\Gamma(r_d) \mapsto (\dots, \Gamma(r_d), \dots)\} \vdash I} \quad \text{(STORE)} \\
 \frac{\Delta; \Gamma\{r_d \mapsto \sigma\}; C; \Sigma \vdash I}{\Delta; \Gamma; C; \Sigma \vdash \text{mov } r_d, r_d; I} \quad \text{(MOVE)} \\
 \frac{\Delta; C - e : \sigma \quad \Delta; \Gamma\{r_d \mapsto \sigma\}; C; \Sigma \vdash I}{\Delta; \Gamma; C; \Sigma \vdash \text{movl } e, r_d; I} \quad \text{(MOVE)} \\
 \frac{\Delta; \Gamma\{r_d \mapsto \Gamma(r_d)\} \{-, -, +\} \Gamma(r_{d1}); C; \Sigma \vdash I}{\Delta; \Gamma; C; \Sigma \vdash (\text{add, sub, mul}) r_d, r_{d1}, r_d; I} \quad \text{(ARITH)} \\
 \frac{\Delta; C \vdash \Gamma(r_d) = \nu; [C'] [\Sigma'] (\Gamma')}{\Delta; C'' \vdash C \quad \Delta; C'' \vdash \Sigma = \Sigma' \quad \Delta; C'' \vdash \Gamma \leq \Gamma'} \quad \text{(BRANCH)} \\
 \frac{\Delta; C \vdash \Gamma(r_d) = \nu; [C'] [\Sigma'] (\Gamma')}{\Delta; C \vdash C' \quad \Delta; C \vdash \Sigma = \Sigma' \quad \Delta; C \vdash \Gamma \leq \Gamma'} \quad \text{(JUMP)} \\
 \Delta; \Gamma; C; \Sigma \vdash \text{jmp } r_d
 \end{array}$$

プログラム検証のための理論の一部
Maeda, T.: Writing an Operating System with a Strictly Typed Assembly Language.
Ph.D. thesis, University of Tokyo (2006) より